

1

Information Based Complexity of Networks

Russell K. Standish

Mathematics and Statistics, University of New South Wales

hpcoder@hpcoders.com.au

1.1

Introduction

Information is a measure of the amount of reduction in uncertainty the receipt of a message causes in the receiver. It is also used interchangeably with the term complexity, referring to a measure of how complex a system might be.

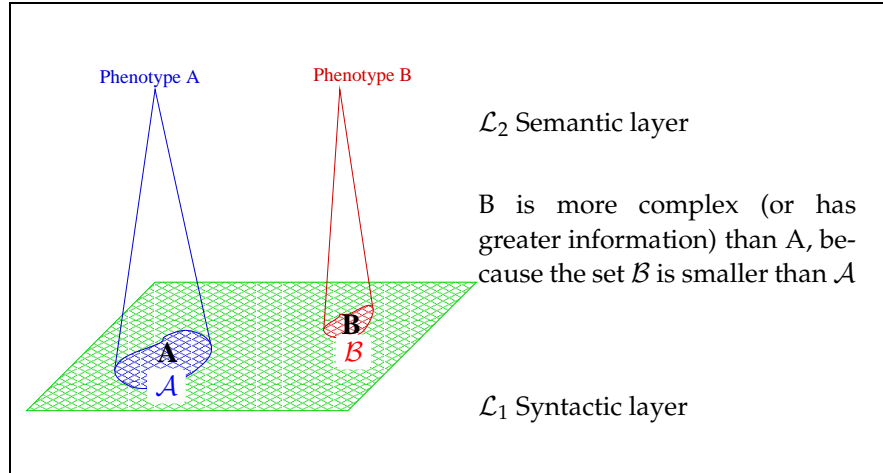
Simple systems with few and regularly behaving parts require little information to describe how the system behaves. Conversely, systems with many similar parts may well admit a simple statistical description that is also of low complexity. Random behaviour, in particular, is of low complexity, as randomness, by definition, entails that no specific model for the behaviour exists, and only simple statistical descriptions are available.

By contrast, a system that needs to be modelled in great detail to capture the essential behaviour, an automobile, or a living cell, is a *complex system*, requiring a large amount of information to specify the systems model.

Note that in the course of the preceding paragraphs, the terms *models* and *descriptions* slipped in. Complexity (and indeed information) is an observer dependent term [36]. What may be simple for the intents and purposes of one observer may well be complex to another. Nevertheless, once a discussion has been adequately framed so that observers agree on what is important about a system being discussed, information theory provides an objective measure of the amount of information or complexity a system exhibits.

When talking about the complexity of networks, it is important to realise that networks in themselves are abstract models of some system. We need to be clear whether the nodes are distinguishable, other than by their position within the network, by labels perhaps, or categories such as colours. There may be dynamics between the parts of the system represented by the network, which needs to be represented in any consideration of complexity.

In what follows, starting with with unlabelled, undirected static networks, we will consider the effects of labelling and colouring nodes, directed edges



Tab. 1.1 Diagram showing the syntactic and semantic spaces. Two different messages, having meanings A and B, can each be coded in many equivalent ways in syntactic space, represented by the sets \mathcal{A} and \mathcal{B} . The information or complexity of the messages is related to the size it occupies in syntactic space by formula (1.1)

between the nodes, weighted edges and finally how to measure the complexity of a dynamical system defined on a network.

1.2

History and concept of information based complexity

Information theory began in the work of Shannon [35], who was concerned with the practical problem of ensuring reliable transmission of messages. Every possible message has a certain probability of occurring. The less likely a message is, the more information it imparts to the listener of that message. The precise relationship is given by a logarithm:

$$I = -\log_2 p \quad (1.1)$$

where p is the probability of the message, and I is the information it contains for the listener. The base of the logarithm determines what units information is measured in — base 2 means the information is expressed in *bits*. Base 256 could be used to express the result in *bytes*, and is of course equivalent to dividing equation (1.1) by 8.

Shannon, of course, was not so interested in the semantic content of the message (ie its meaning), rather in the task of information transmission so instead considered a message composed of symbols x_i drawn from an alphabet A . Each symbol had a certain probability $p(x_i)$ of appearing in a message —

consider how the letter ‘e’ is far more probable in English text than the letter ‘q’. These probabilities can be easily measured by examining extant texts. A first order approximation to equation (1.1) is given by:

$$I(x_1 x_2 \dots x_n) \approx \sum_{i=1}^n p(x_i) \log_2 p(x_i) \quad (1.2)$$

This equation can be refined by considering possible pairs of letters, then possible triplets, in the limit converging on the minimum amount of information required to be transmitted in order for the message to be reconstructed in its original form. That this value may be considerably less than just sending the original message in its entirety is the basis of compression algorithms, such as those employed by the well-known *gzip* or *PKzip* (aka WinZip) programs.

The issue of semantic content discouraged a lot of people from applying this formalism to complexity measures. The problem is that a message written in English will mean something to a native English speaker, but be total gibberish to someone brought up in the Amazon jungle with no contact with the English speaking world. The information content of the message depends on exactly who the listener is! Whilst this context dependence appears to make the whole enterprise hopeless, it is in fact a feature of all the naive complexity measures normally discussed. When counting the number of parts in a system, one must make a decision as to what exactly constitutes a part, which is invariably somewhat subjective, and needs to be decided by consensus or convention by the parties involved in the discussion. Think of the problems in trying decide whether a group of animals is one species or two, or which genus they belong to. The same issue arises with the characterisation of the system by a network. When is a relationship considered a graph edge, when often every component is connected to every other part in varying degrees.

However, in many situations, there appears to be an obvious way of partitioning the system, or categorising it. In such a case, where two observers agree on the same way of interpreting a system, then they can agree on the complexity that system has. If there is no agreement on how to perform this categorisation, then complexity is meaningless.

To formalise complexity then, assume as given a classifier system that can categorise descriptions into equivalence classes. This is sketched in Figure 1.1, where sets of descriptions in the syntactic layer \mathcal{L}_1 are mapped to messages in the semantic layer \mathcal{L}_2 . Clearly, humans are very good at this — they’re able to recognise patterns even in almost completely random data. Rorschach plots are random ink plots that are interpreted by viewers as a variety of meaningful images. However, a human classifier system is not the only possibility. Another is the classification of programs executed by a computer by what output they produce. Technically, in these discussions, researchers use a *Universal Turing Machine* (UTM), an abstract model of a computer.

Consider then the set of possible binary strings, which can be fed into a UTM U as a program. Some of these programs cause U to produce some output then halt. Others will continue executing forever. In principle, it is impossible to determine generally if a program will halt or continue on indefinitely. This is the so called *halting problem*. Now consider a program p that causes the UTM to output a specific string s and then halt. Since the UTM halts after a certain number of instructions executed (denoted $\ell(p)$) the same result is produced by feeding in any string starting with the same $\ell(p)$ bits. If the strings have equal chance of being chosen (*uniform measure*), then the proportion of strings starting with the same initial $\ell(p)$ bits is $2^{-\ell(p)}$. This leads to the *universal prior* distribution over descriptions s , also known as the Solomonoff-Levin distribution:

$$P(s) = \sum_{\{p:U(p)=s\}} 2^{-\ell(p)} \quad (1.3)$$

The complexity (or information content) of the description is given by equation (1.1), or simply the logarithm of (1.3). In the case of an arbitrary classifier system, the complexity is given by the negative logarithm of the equivalence class size

$$\mathcal{C}(x) = \lim_{s \rightarrow \infty} s \log_2 N - \log_2 \omega(s, x) \quad (1.4)$$

where N is the size of the alphabet used to encode the description and $\omega(s, x)$ is the number of equivalent descriptions having meaning x of size s or less [36].

It turns out that the probability $P(s)$ in equation (1.3) is dominated by the shortest program [24, Thm 4.3.3], namely

$$K(s) + \log_2 P(s) \leq C \quad (1.5)$$

($\log_2 P(s) < 0$ naturally) where C is a constant independent of the description s . $K(s)$ is the length of the shortest program p that causes U to output s , and is called the *Kolmogorov complexity* or *algorithmic complexity*.

An interesting difference between algorithmic complexity, and the general complexity based on human observers can be seen by considering the case of random strings. *Random*, as used in algorithmic information theory, means that no shorter algorithm can be found to produce a string than simply saying “print ...”, where the ... is a literal representation of the string. The algorithmic complexity of a random string is high, at least as high as the length of the string itself. However, a human observer simply sees a random string as a jumble of letters, much the same as any other random string. In this latter case, the equivalence class of random strings is very large, close to N^s , so the perceived complexity is small. Thus the human classifier defines an example of what Gell-Mann calls *effective complexity* [16], namely a complexity that

has a high value for descriptions that are partially compressible by complex schema, but low for random or obviously regular systems.

A good introduction to information theoretical concepts for complex systems studies can be found in [2].

1.3

Mutual Information

When considering information *transfer*, it is useful to consider the amount of information transferred in a message to be related to the reduction in uncertainty the receiver has about the source on receipt of the message. In order to quantify this, consider the sender and receiver to be stochastic variables X and Y , and form the joint probability:

$$P(X = x_i \text{ and } Y = y_j) = p(x_i, y_j) \quad (1.6)$$

We can then form the entropies

$$\begin{aligned} H(X) &= \sum_i P(X = x_i) \log P(X = x_i) \\ H(Y) &= \sum_i P(Y = y_i) \log P(Y = y_i) \end{aligned}$$

and the joint entropy

$$H(X, Y) = \sum_{ij} p(x_i, y_j) \log p(x_i, y_j).$$

If the processes X and Y are independent of each other, we have

$$p(x_i, y_j) = P(X = x_i)P(Y = y_j),$$

so therefore

$$H(X, Y) = H(X) + H(Y)$$

for independent processes. In general, however

$$H(X, Y) \leq H(X) + H(Y).$$

The difference is known as *mutual information*:

$$I(X : Y) = H(X) + H(Y) - H(X, Y) \quad (1.7)$$

Conditional entropy is the usual entropy applied to conditional probability $P(X = x_i | y_j)$:

$$H(X|Y) = \sum_{ij} p(x_i, y_j) \log P(X = x_i | y_j). \quad (1.8)$$

Using Bayes rule, mutual information can be expressed in terms of the conditional entropy as

$$I(X : Y) = H(X) - H(X|Y) = H(Y) - H(Y|X). \quad (1.9)$$

1.4

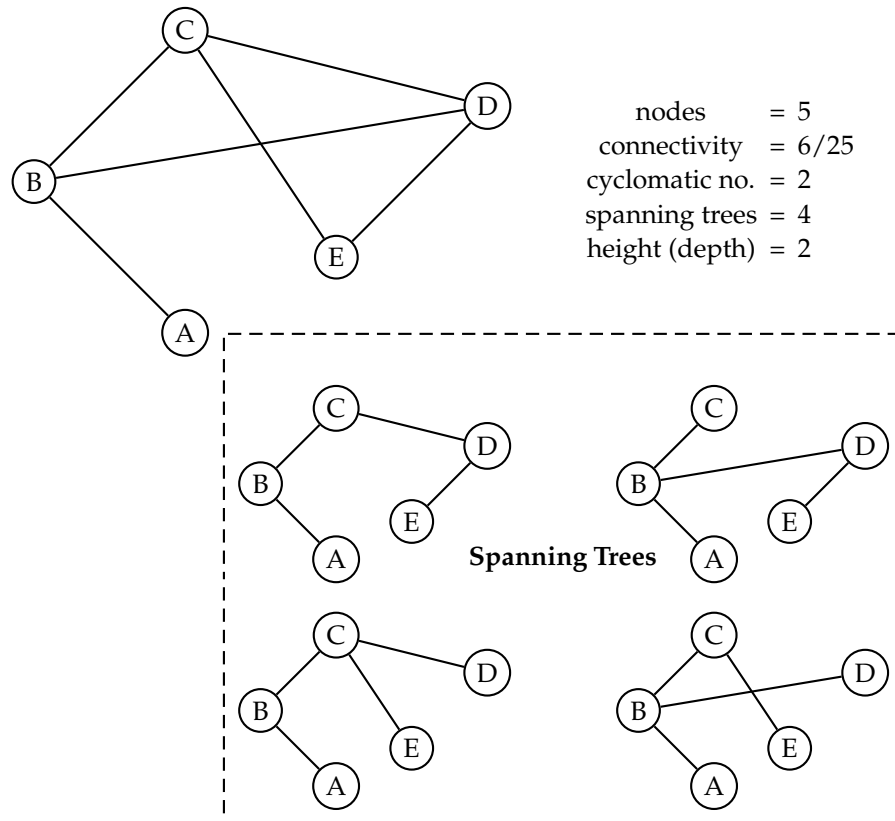
Graph theory, and graph theoretic measures: cyclomatic number, spanning trees

Systems with many similar, barely interacting parts are clearly quite simple. Contrasting a pile of sand with a silicon chip, we naturally want our complexity measure to capture the inherent complexity in the silicon chip, even if they're made of similar numbers of parts of similar material.

Since the pile of sand case indicates complexity is not simply the number of components making up a system, the relationships between components clearly contribute to the overall complexity. One can start by caricaturing the system as a *graph* — replacing the components by abstract *vertices* or *nodes* and relationships between nodes by abstract *edges* or *arcs*.

Graph theory [13] was founded by Euler in the 18th century to solve the famous Königsberg bridge problem. However, until the 1950s, only simple graphs that could be analysed in toto were considered. Erdős and Renyi [15] introduced the concept of a *random graph*, which allowed one to treat large complex graphs statistically. Graphs of various sorts were readily recognised in nature, from food webs, personal or business contacts, sexual relations and the Internet amongst others. However, it soon became apparent that natural networks often had different statistical properties than general random graphs. Watts and Strogatz [45] introduced the *small world* model, which has sparked a flurry of activity in recent years to measure networks such as the Internet, networks of collaborations between scientific authors and food webs in ecosystems [3].

Graph theory provides a number of measures that can stand in for complexity. A number of these are illustrated in Figure 1.2. The simplest of these is the *connectivity* of a graph, namely the number of edges connecting vertices of the graph. A fully connected graph, however, is no more complex than one that is completely unconnected. As connectivity increases from zero, a *percolation threshold* is reached where the graph changes from being mostly discontinuous to mostly continuous. The most complex systems tend to lie close to the percolation threshold. Another graph measure used is the *cyclomatic number* of a graph, basically the number of independent loops it contains. The justification for using cyclomatic number as a measure of complexity is that feedback loops introduce nonlinearities in the system's behaviour, that produce complex behaviour.



Tab. 1.2 Various graph theoretic measures for a simple graph. The spanning trees are shown in the dashed box

Related to the concept of cyclomatic number is the *number of spanning trees* of the graph. A spanning tree is a subset of the graph that visits all nodes but has no loops (ie is a tree). A graph made up from several disconnected parts has no spanning tree. A tree has exactly one spanning tree. The number of spanning trees increases rapidly with the cyclomatic number.

The height of the flattest spanning tree, or equivalently the maximum number of hops separating two nodes on the graph is another useful measure related to complexity, usually called the *diameter*. Networks having small degrees of separation (so called *small world networks*) tend to support more complex dynamics than networks having a large degree of separation. The reason is that any local disturbance is propagated a long way through a small world network before dying out, giving rise to chaotic dynamics, whereas in the

other networks, disturbances remain local, leading to simpler linear dynamics.

1.5

Erdos-Renyi random graphs, small world networks, scale-free networks

When considering the statistical properties of large networks, it is useful to randomly generate networks having particular properties from simple models. These may be used, for instance, as null models, to determine if the network being studied has attributes that are statistically significantly different from the null model.

The simplest such random model was introduced in the 1950s by Erdős and Rényi [15]. Starting with n nodes, add ℓ edges by randomly selecting pairs of nodes and attaching an edge. Equivalently, one can add an edge between any pair of nodes with probability $p = \ell/n(n-1)$. Erdős-Rényi graphs exhibit a Gaussian degree distribution, and substantially more clustering compared with graphs embedded in a low dimensional space (eg wireframe meshes).

Graphs embedded in a low dimensional space¹ have a high graph diameter (many edges need to be traversed to pass from one node to another randomly chosen node). By contrast, random graphs of sufficiently high connectivity tend to have low diameter, between any randomly chosen pair of nodes, there will be a path traversing only a few edges, a property called *small world*. One can construct small world graphs in between Cartesian graphs and random graphs by starting with a Cartesian graph, and randomly rewiring a small proportion of the edges.

Many real world networks exhibit a *scale free* property, with the node degree distribution following a power law. One popular algorithm for generating these sorts of networks is *preferential attachment*, which involves adding links preferentially to nodes with higher degree in a “rich gets richer” effect [7].

1.6

Graph entropy

There is a long tradition of applying information theory to graph structures, starting with Rashevsky [33], Trucco [40] and Mowshowitz [27–30]. A recent, detailed review can be found in [12].

¹) the regular ones are usually called Cartesian graphs, as the nodes are just the points whose Cartesian coordinates are integral

Given a graph $G = V \times E$ of nodes V and links E , and *graph invariant* function α defined on the nodes:

$$\alpha : V \rightarrow A$$

we can form the graph entropy measure

$$S(G, \alpha) = |V| \log |V| - \sum_{a \in A} |\alpha^{-1}(a)| \log |\alpha^{-1}(a)|, \quad (1.10)$$

where $|\cdot|$ is the usual notation for set cardinality. The sum in eq (1.10) is over sets of nodes that are equivalent under the map α . This plays the analogous role to the observer function $O(x)$ mentioned previously. In §1.7, we will use the automorphism relation between graphs as the observer function — the corresponding α function maps nodes to their *orbits*. Other graph invariants have also been used in the literature, such as node degree, or level in a tree structure.

A very similar measure to (1.10) is obtained by averaging the information contained in each orbit:

$$I(G, \alpha) = - \sum_{a \in A} P_i \log P_i = - \sum_{a \in A} \frac{|\alpha^{-1}(a)|}{|V|} \log \frac{|\alpha^{-1}(a)|}{|V|} \quad (1.11)$$

One can likewise form similar measures by considering graph invariants over links, rather than nodes.

1.7

Information based complexity of unweighted, unlabeled, undirected networks

In order to compute the complexity according to equation (1.4), it is necessary to fix two things: a bitstring representation (*description*) of the item in question, and a means of determining if two descriptions describe the same object.

In the case of graphs, we consider two graphs to be identical if and only if a permutation of nodes exists that allows the nodes of one graph to be placed in a 1–1 correspondence with the nodes of the other. In other words, an *automorphism*. If either the nodes or edges are labelled, or a dynamic process is defined on the network, a situation we will consider in subsequent sections, then the labelling (or process in that case) must also be preserved by the automorphism.

One very simple implementation language for undirected graphs is to label the nodes $1 \dots n$, and the links by the pair (i, j) , $i < j$ of nodes that the links connect. The linklist can be represented simply by an $L = n(n - 1)/2$ length

bitstring, where the $\frac{1}{2}j(j-1) + i$ th position is 1 if link (i, j) is present, and 0 otherwise.

The directed case requires doubling the size of the linklist, ie or $L = n(n-1)$. We also need to prepend the string with the value of N in order to make it prefix-free — the simplest approach being to interpret the number of leading 1s as the number n , which adds a term $n+1$ to the measured complexity.

This proposal was analysed in [37], and has the unsatisfactory property that the fully connected or empty networks are maximally complex for a given node count. An alternative scheme is to also include the link count as part of the prefix, and to use binary coding for both the node and link counts [38]. The sequence will start with $\lceil \log_2 n \rceil$ 1's, followed by a zero stop bit, so the prefix will be $2\lceil \log_2 n \rceil + \lceil \log_2 L \rceil + 1$ bits.

This scheme entails that some of bitstrings are not valid networks, namely ones where the link count does not match the number of 1s in the linklist. We can, however, use rank encoding [31] of the linklist to represent the link pattern. The number of possible linklists corresponding to a given node/link specification is given by

$$\Omega = \binom{L}{l} = \frac{L!}{(L-l)!l!} \quad (1.12)$$

This will have a minimum value of 1 at $l = 0$ (empty network) and $l = L$, the fully connected network.

Finally, we need to compute ω of the linklist, which is just the total number of possible renumberings of the nodes ($n!$), divided by the size of the graph automorphism group $|\mathcal{A}|$, which can be practically computed by Nauty [26], or a number of other algorithms which exhibit better performance on sparsely linked networks [10, 11, 20]. With ω computed, the complexity \mathcal{C} of the network is given by (1.4).

A network A that has a link wherever B doesn't, and vice-versa might be called a complement of B . A bitstring for A can be found by inverting the 1s and 0s in the linklist part of the network description. Obviously, $\omega(A, L) = \omega(B, L)$, so the complexity of a network is equal to that of its complement, as can be seen in Figure 1.3.

A connection between \mathcal{C} and the graph entropy S defined in equation (1.10) can be made by noting that the size of the automorphism group is simply the product of the sizes of the orbits:

$$|\mathcal{A}| = \prod_{a \in A} |\alpha^{-1}(a)|! \quad (1.13)$$

Tab. 1.3 The new complexity measure as a function of link count for all networks with 8 nodes. This shows the strong dependence of complexity on link count, and the symmetry between networks and their complements.

Using the Stirling approximation ($\log x! \approx x \log x$), we may write

$$\begin{aligned} \omega &= \frac{n!}{|\mathcal{A}|} = \frac{|V|!}{\prod_{a \in A} |\alpha^{-1}(a)|!} \\ \log \omega &\approx |V| \log |V| - \sum_{a \in A} |\alpha^{-1}(a)| \log |\alpha^{-1}(a)| = S \end{aligned} \quad (1.14)$$

1.8

Motif expansion

Adami et al. [1] introduce the concept of *motif entropy*. By breaking the network into motifs (eg pairs of nodes connected by a link, triangles, quads, 3-pointed stars, etc), and forming the Shannon entropy $H = -\sum_i p_i \log_2 p_i$, where p_i are the probabilities of the various motifs occurring, one gets a measure which they call motif entropy. This should converge to (1.4) as more motifs are included, in just the same way as (1.2) converges to (1.1) as longer sequences are included. Adami et al. restrict themselves to motifs of two and three nodes only in examining the neural network of *C. elegans*, and also in ex-

aming the epistatic interaction networks in the Avida digital organism system. They show that this suffices to capture meaningful adaptive information about the systems, but not whether it captures all the pertinent information. Further work linking motif entropy with network complexity is called for.

1.9

Labelled networks

If all nodes are labelled with distinct labels, the network is uniquely specified by the node and link counts, along with a rank-encoded linklist. In this case, eq (1.4) can be expressed analytically:

$$\mathcal{C} = 2\lceil \log_2 n \rceil + \lceil \log_2 L \rceil + 1 + \left\lceil \log_2 \frac{L!}{(L-l)!!} \right\rceil \quad (1.15)$$

In the case where the labels are not distinct, the network is often said to be *coloured*². Not much work has been done calculating the complexity of coloured networks, but recently Adami et al. [1] tackled the problem. In that paper, motif expansion was used to approximate the complexity.

One can use eq (1.4) directly, provided one had an algorithm for computing the size of the automorphism group that leaves the colour labels invariant. This is still an open problem, but in principle, existing automorphism algorithms should be able to be adapted.

However, special cases exist where the coloured network complexity reduces to uncoloured network complexity. For example, if all nodes within a colour grouping have distinct degree, then the problem is identical to the distinct label case, and equation (1.15) can be used. Similarly, if all nodes of the same degree have the same colour, then the coloured network complexity is identical to that of the uncoloured network.

1.10

Weighted networks

Whilst the information contained in link weights might be significant in some circumstances (for instance the weights of a neural network can only be varied in a limited range without changing the overall qualitative behaviour of the network), of particular theoretical interest is to consider the weights as continuous parameters connecting one network structure with another. For instance if a network X has the same network structure as the unweighted graph A , with b links of weight 1 describing the graph B and the remaining

2) Eg. if there are three distinct labels, they may as well be red, green and blue.

$a - b$ links of weight w , then we would like the network complexity of X to vary smoothly between that of A and B as w varies from 1 to 0. [17] introduced a similar measure.

The most obvious way of defining this continuous complexity measure is to start with normalised weights $\sum_i w_i = 1$. Then arrange the links in weight order, and compute the complexity of networks with just those links of weights less than w . The final complexity value is obtained by integrating:

$$\mathcal{C}(X = N \times L) = \int_0^1 \mathcal{C}(N \times \{i \in L : w_i < w\}) dw \quad (1.16)$$

Obviously, since the integrand is a stepped function, this is computed in practice by a sum of complexities of partial networks.

1.11

Empirical results of real network data, and artificially generated networks

Table 1.4 shows the complexities of a number of well-known real world networks [38]. Also shown is the average complexity of 1000 shuffled networks. Shuffling the links of a network produces an Erdős-Rényi random network with an identical link weight distribution to the original network.

In most cases, there is a statistically significant difference between the real network complexity and the shuffled version, indicating that the network structure encodes significant information.

In [38], several evolutionary systems from Artificial Life are also analysed in the same way, as well as networks generated by the Erdős-Rényi process and the Barabási-Albert *preferential attachment process* [7]. Networks derived from evolutionary process exhibited the same sort of complexity excess as the real world network, but networks created from purely random processes did not, indicating the information hoarding nature of adaption.

1.12

Extension to processes on networks

What has been discussed up until now is the static, or *structural* complexity of a network. Often, a dynamic process occurs on a network, such as neural network dynamics, or the ecological dynamics of a foodweb. One is interested in the amount of complexity contributed to the process by the network structure. Since two distinct networks with the same attached dynamics may well be considered identical in some context (perhaps by having the same attractor basins, for instance), then in general the *dynamic complexity* is less than the *structural complexity*.

Dataset	nodes	links	\mathcal{C}	$e^{\langle \ln \mathcal{C}_{\text{ER}} \rangle}$	$\mathcal{C} - e^{\langle \ln \mathcal{C}_{\text{ER}} \rangle}$	$\frac{ \ln \mathcal{C} - \langle \ln \mathcal{C}_{\text{ER}} \rangle }{\sigma_{\text{ER}}}$
celegansneural	297	2345	442.7	251.6	191.1	29
lesmis	77	508	199.7	114.2	85.4	24
adjnoun	112	850	3891	3890	0.98	∞
yeast	2112	4406	33500.6	30218.2	3282.4	113.0
celegansmetabolic	453	4050	25421.8	25387.2	34.6	∞
baydry	128	2138	126.6	54.2	72.3	22
baywet	128	2107	128.3	51.0	77.3	20
cypdry	71	641	85.7	44.1	41.5	13
cypwet	71	632	87.4	42.3	45.0	14
gramdry	69	911	47.4	31.6	15.8	10
gramwet	69	912	54.5	32.7	21.8	12
Chesapeake	39	177	66.8	45.7	21.1	10.4
ChesLower	37	178	82.1	62.5	19.6	10.6
ChesMiddle	37	208	65.2	48.0	17.3	9.3
ChesUpper	37	215	81.8	60.7	21.1	10.2
CrystalC	24	126	31.1	24.2	6.9	6.4
CrystalD	24	100	31.3	24.2	7.0	6.2
Everglades	69	912	54.5	32.7	21.8	11.8
Florida	128	2107	128.4	51.0	77.3	20.1
Maspalomas	24	83	70.3	61.7	8.6	5.3
Michigan	39	219	47.6	33.7	14.0	9.5
Mondego	46	393	45.2	32.2	13.0	10.0
Narragan	35	219	58.2	39.6	18.6	11.0
Rhode	19	54	36.3	30.3	6.0	5.3
StMarks	54	354	110.8	73.6	37.2	16.0
PA1	100	99	98.9	85.4	13.5	2.5
PA3	100	177	225.9	207.3	18.6	3.0

Tab. 1.4 Complexity values of several freely available network datasets. celegansneural, lesmis and adjnoun are available from Mark Newman’s website, representing the neural network of the *C. elegans* nematode [45], the coappearance of characters in the novel *Les Misérables* by Victor Hugo [23] and the adjacency network of common adjectives and nouns in the novel *David Copperfield* by Charles Dickens [32]. The metabolic data of *C. elegans* [14] and protein interaction network in yeast [19] are available from Duncan Watt’s website. PA1 and PA3 are networks generated via preferential attachment with in degree of one or three respectively, and uniformly distributed link weights. The other datasets are food webs available from the Pajek website [4–6, 9, 18, 42, 43]. For each network, the number of nodes and links are given, along with the computed complexity \mathcal{C} . In the fourth column, the original network is shuffled 1000 times, and the logarithm of the complexity is averaged ($\langle \ln \mathcal{C}_{\text{ER}} \rangle$). The fifth column gives the difference between these two values, which represents the information content of the specific arrangement of links. The final column gives a measure of the significance of this difference in terms of the number of standard deviations (“sigmas”) of the distribution of shuffled networks. In two examples, the distributions of shuffled networks had zero standard deviation, so ∞ appears in this column.

With continuous processes, there is a practical difficulty of establishing whether two networks generate the same process, particularly if there is an element of stochasticity involved. When comparing continuous-valued time series, one would need to choose a metric over (in general) a multidimensional

space, and an error threshold within which two time series are considered the same. Furthermore, a maximum time period for comparison needs to be chosen, as dynamical chaos effects are likely to render two arbitrarily close trajectories significantly different after a finite period of time.

If the initial transients of the processes aren't important, one could compare basins of attraction instead, which only eliminates the choice of time period in the comparison.

With discrete (or symbolic) processes, the problem is conceptually simpler in that one can determine if two networks generate identical processes according to an observer function $O(x)$.

Nevertheless, the computational complexity of this approach rules it out for all but the simplest of networks.

An alternative approach is given by considering the amount of information flowing between nodes, a notion known as *transfer entropy*.

1.13

Transfer Entropy

Given a time series X_t , let $X_t^- = \{X_t, X_{t-1}, X_{t-2}, \dots\}$ be the *history* of X up to time t .

The mutual information $I(X_{t+1} : Y_t^-)$ gives a measure of the extent to which the history of Y disambiguates the future of X . However, Y may itself depend on the past of X , giving rise to spurious directional effects between Y and X [21]. So we should also condition on the past of X , giving rise to the notion of *transfer entropy*:

$$\mathcal{T}_{Y \rightarrow X} = I(X_{t+1} : Y_t^- | X_t^-) \quad (1.17)$$

Transfer entropy has been applied to random boolean networks [25], but is more usually used to infer network structure from time series data such as neural networks [46] or genetic regulatory networks [41].

We may also condition the transfer entropy on the state of the rest of system U (not including X or Y):

$$\mathcal{T}_{Y \rightarrow X|U} = I(X_{t+1} : Y_t^- | X_t^-, U_t^-). \quad (1.18)$$

Lizier et al [25] call (1.17) the *apparent* transfer entropy, and (1.18) the *complete* transfer entropy. They find that apparent transfer entropy is maximised around a critical point corresponding to a connectivity of around 2 links per node, whereas complete transfer entropy rises near the critical point, and continues to rise as connectivity increases and the system moves into the chaotic regime, up to a connectivity of 5 links per node.

A related concept to transfer entropy is *Granger causality*. Granger causality between two nodes X and Y is found by considering a linear multivariate model of the lags

$$X_{t+1} = \sum_k A_k X_{t-k} + \sum_k B_k Y_{t-k} + \epsilon$$

and a restricted linear model with the Y terms removed:

$$X_{t+1} = \sum_k C_k X_{t-k} + \epsilon'.$$

If the former full model gives a statistically significant better fit to the data than the latter restricted model, we say that Y *Granger causes* X . To quantify the statistical significance, we use the F-statistic

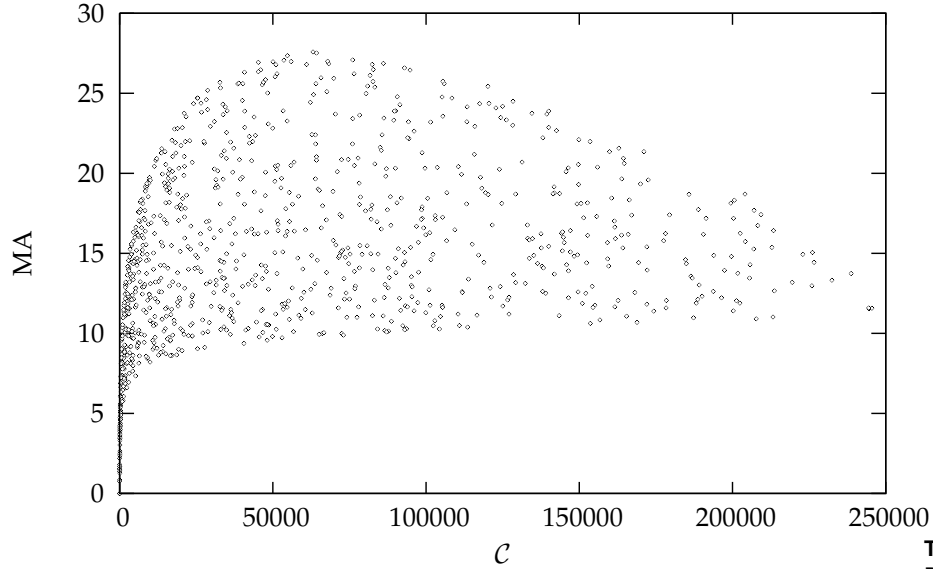
$$\mathcal{F}_{Y \rightarrow X} = \ln \frac{\langle \epsilon'^2 \rangle}{\langle \epsilon^2 \rangle} \quad (1.19)$$

as the variance of residuals in the restricted model ($\langle \epsilon'^2 \rangle$) will be more than that of the full model.

Granger causality has the advantages of being computationally simpler, as well as having an interpretation in terms of statistical significance. The downside is that it captures linear relationships only, whereas transfer entropy is model-free, capturing all that is relevant between entities. The two concepts are very closely related, and for the special case of Gaussian processes, are identical up to a factor of 2 [8].

The models used in Granger causality may also include the remainder of the system U , and this is used for computing the *causal density* of the system, which is the proportion of pairs of nodes where one node *Granger causes* the other [34]. The measure has a minimum for weakly interacting nodes, and likewise for strongly interacting nodes (as everything influences everything else, so is conditioned out). It has a maximum in between, expressing a balance between integration and segregation in a system. It is very similar to an earlier measure proposed by Tononi, Sporns and Edelman (TSE complexity) [39] which is based on mutual information across bipartitions of the network rather than transfer entropy.

Like \mathcal{C} from (1.16), both TSE complexity and causal density are minimal for sparse and dense networks, rising to a maximum value in between. However the maximum value of causal density occurs around the order-chaos transition (approx 2 links per node), which is a distinctly different peak to that of structural complexity, which is at a maximum at $n/2$ links per node.



Tab. 1.5 Medium Articulation of Erdős-Rényi graphs up to order 500.

1.14

Medium Articulation

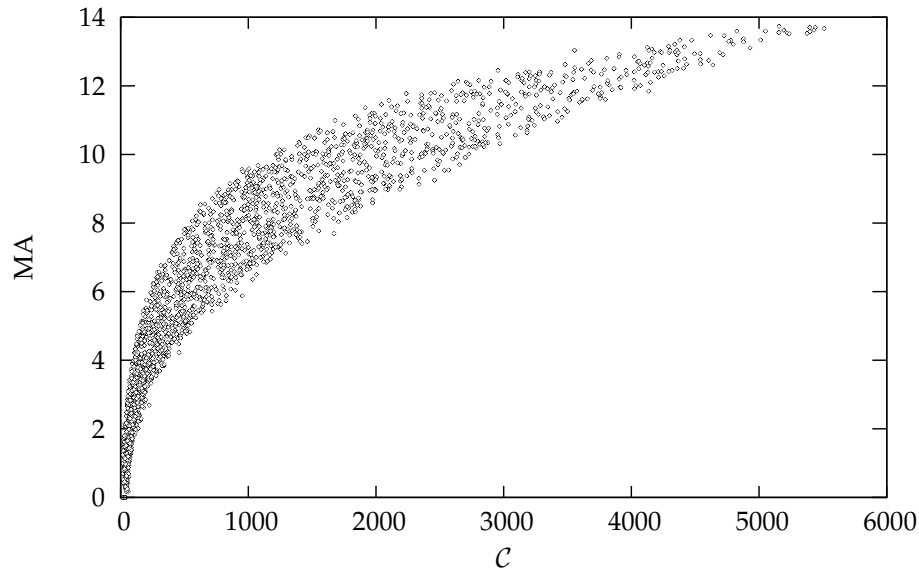
Wilhelm [22,47] introduced a new complexity like measure that addresses the intuition that complexity should be minimal for the empty and full networks, and peak for intermediate values (like figure 1.3). It is obtained by multiplying the mutual information between all pairs of nodes by the conditional entropy across all links (which they call the *redundancy*). The resulting measure also has a quality of measuring the segregation/integration balance reminiscent of causal density.

Precisely, medium articulation is given by

$$\text{MA} = - \sum_{ij} w_{ij} \log \frac{w_{ij}}{\sum_k w_{ik} \sum_k w_{kj}} \times \sum_{ij} w_{ij} \log \frac{w_{ij}^2}{\sum_k w_{ik} \sum_k w_{kj}}, \quad (1.20)$$

where w_{ij} is the normalised weight ($\sum_{ij} w_{ij} = 1$) of the link from node i to node j . It should be noted that this is just the product of the two terms A and Φ representing the degree of constraint and the extent of freedom of the system in Ulanowicz's paper in this volume [44].

Figure 1.5 shows medium articulation plotted against C for a sample of 1000 Erdős-Rényi networks up to order 500. There is no clear relationship between medium articulation and complexity for the average network. Medium articulation does not appear to discriminate between complex networks. However if we restrict our attention to simple networks (Figures 1.6 and 1.7) medium



Tab. 1.6 Medium Articulation
Rényi graphs up to order 500

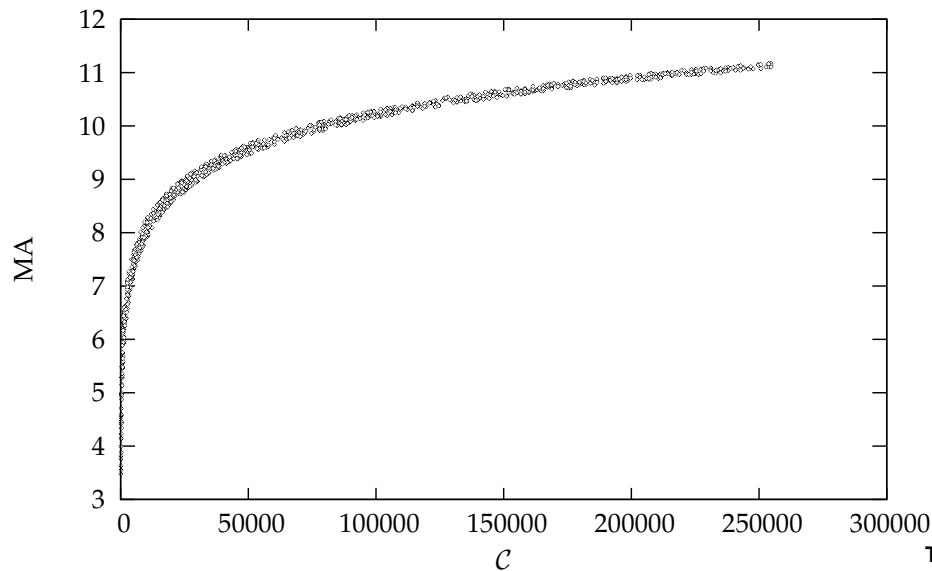
articulation is strongly correlated with complexity, and so could be used as a proxy for complexity for these cases.

This lends some credence to the notion that causal density, TSE complexity and network complexity are all related.

1.15

Conclusion

In this chapter, a number of information-based measures of network complexity are considered. Measures of structural complexity are found to be related to each other, and similarly information flow measures of dynamic complexity are also found to be related. It would seem plausible that dynamic complexity measures should be related to structural complexity when the dynamical processes are in some sense generic, or uncoloured, but at this stage, such a conjecture remains unproven. For relatively simple processes such as Gaussian processes, and the Random Boolean Networks studied by Lizier et al., the behaviour of a dynamical complexity measure has a peak at much lower connectivities than the peak exhibited by the structural complexity measure. More work is required to clarify the relationship between dynamical and structural complexity of networks.



Tab. 1.7 Medium Articulation p Rényi graphs up to order 500 v

Bibliography

- 1 C. Adami, J. Qian, M. Rupp, and A. Hintze. Information content of colored motifs in complex networks. *Artificial Life*, 17:375–390, 2011.
- 2 Chris Adami. *Introduction to Artificial Life*. Springer, 1998.
- 3 Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74:47, 2002.
- 4 J. Almunia, G. Basterretxea, J. Aristegui, and R.E. Ulanowicz. Benthic- pelagic switching in a coastal subtropical lagoon. *Estuarine, Coastal and Shelf Science*, 49:363–384, 1999.
- 5 D. Baird, J. Luczkovich, and R. R. Christian. Assessment of spatial and temporal variability in ecosystem attributes of the St Marks National Wildlife Refuge, Apalachee Bay, Florida. *Estuarine, Coastal, and Shelf Science*, 47:329–349, 1998.
- 6 D. Baird and R.E. Ulanowicz. The seasonal dynamics of the Chesapeake Bay ecosystem. *Ecological Monographs*, 59:329–364, 1989.
- 7 Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.
- 8 L. Barnett, A.B. Barrett, and A.K. Seth. Granger causality and transfer entropy are equivalent for gaussian variables. *Physical review letters*, 103(23):238701, 2009.
- 9 R.R. Christian and J.J. Luczkovich. Organizing and understanding a winter’s seagrass foodweb network through effective trophic levels. *Ecological Modelling*, 117:99–124, 1999.
- 10 L.P. Cordella, P. Foggia, C. Sansone, and M. Vento. An improved algorithm for matching large graphs. In *3rd LAPR-TC15 workshop on graph-based representations in pattern recognition*, pages 149–159, 2001.
- 11 P. T. Darga, K. A. Sakallah, and I. L. Markov. Faster symmetry discovery using sparsity of symmetries. In *Proceedings of the 45st Design Automation Conference*, Anaheim, California, June 2008.
- 12 M. Dehmer and A. Mowshowitz. A history of graph entropy measures. *Information Sciences*, 181(1):57–78, 2011.

- 13 Reinhard Diestel. *Graph Theory*. Springer, Berlin, 3rd edition, 2005.
- 14 J. Duch and A. Arenas. Community identification using extremal optimization. *Physical Review E*, 72:027104, 2005.
- 15 Paul Erdős and Alfréd Rényi. On random graphs. *Publ. Math. Debrecen*, 6:290–291, 1959.
- 16 Murray Gell-Mann. *The Quark and the Jaguar: Adventures in the Simple and the Complex*. Freeman, 1994.
- 17 Olof Görnerup and James P. Crutchfield. Hierarchical self-organization in the finitary process soup. *Artificial Life*, 14:245–254, 2008.
- 18 J.D. Hagy. *Eutrophication, hypoxia and trophic transfer efficiency in Chesapeake Bay*. PhD thesis, University of Maryland at College Park (USA), 2002.
- 19 Hawoong Jeong, Sean Mason, Albert-László Barabási, and Zoltán N. Oltvai. Centrality and lethality of protein networks. *Nature*, 411:41, 2001.
- 20 Tommi Junttila and Petteri Kaski. Engineering an efficient canonical labeling tool for large and sparse graphs. In *Proceedings of the Ninth Workshop on Algorithm Engineering and Experiments (ALENEX07)*, pages 135–149. SIAM, 2007.
- 21 A. Kaiser and T. Schreiber. Information transfer in continuous processes. *Physica D: Nonlinear Phenomena*, 166:43–62, 2002.
- 22 Jongwang Kim and Thomas Wilhelm. What is a complex graph. *Physica A*, 387:2637–2652, 2008.
- 23 D. E. Knuth. *The Stanford GraphBase: A Platform for Combinatorial Computing*. Addison-Wesley, Reading, MA, 1993.
- 24 Ming Li and Paul Vitányi. *An Introduction to Kolmogorov Complexity and its Applications*. Springer, New York, 2nd edition, 1997.
- 25 Joseph Lizier, Siddharth Pritam, and Mikhail Prokopenko. Information dynamics in small-world boolean networks. *Artificial Life*, 17:293–314, 2011.
- 26 Brendan D. McKay. Practical graph isomorphism. *Congressus Numerantium*, 30:45–87, 1981.
- 27 A. Mowshowitz. Entropy and the complexity of graphs: I. an index of the relative complexity of a graph. *Bulletin of Mathematical Biology*, 30(1):175–204, 1968.
- 28 A. Mowshowitz. Entropy and the complexity of graphs: II. the information content of digraphs and infinite graphs. *Bulletin of Mathematical Biology*, 30(2):225–240, 1968.
- 29 A. Mowshowitz. Entropy and the complexity of graphs: III. graphs with prescribed information content. *Bulletin of Mathematical Biology*, 30(3):387–414, 1968.
- 30 A. Mowshowitz. Entropy and the complexity of graphs: IV. entropy measures and graphical structure. *Bulletin of Mathematical Biology*, 30(4):533–546, 1968.
- 31 Wendy Myrvold and Frank Ruskey. Ranking and unranking permutations in linear time. *Information Processing Letters*, 79:281–284, 2001.
- 32 M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E*, 74:036104, 2006.
- 33 N. Rashevsky. Life, information theory, and topology. *Bulletin of Mathematical Biology*, 17(3):229–235, 1955.
- 34 Anil K. Seth, Adam B. Barrett, and Lionel Barnett. Causal density and integrated information as measures of conscious level. *Phil. Trans. R. Soc. A*, 369:3748–3767, 2011.
- 35 Claude E. Shannon. *The Mathematical Theory of Communication*. U. Illinois Press, Urbana-Champaign, IL, 1949.
- 36 Russell K. Standish. On complexity and emergence. *Complexity International*, 9, 2001. arXiv:nlin.AO/0101006.
- 37 Russell K. Standish. Complexity of networks. In Abbass et al., editors, *Recent Advances in Artificial Life*, volume 3 of *Advances in Natural Computation*, pages 253–263, Singapore, 2005. World Scientific. arXiv:cs.IT/0508075.
- 38 Russell K. Standish. Complexity of networks (reprise). *Complexity*, 17:50–61, 2012. arXiv: 0911.348.
- 39 G. Tononi, O. Sporns, and G. M. Edelman. A measure for brain complexity: relating functional segregation and integration in

- the nervous system. *Proc. Natl Acad. Sci. USA*, 91:5033–5037, 1994.
- 40** E. Trucco. A note on the information content of graphs. *Bulletin of Mathematical Biology*, 18(2):129–135, 1956.
- 41** Thai Quang Tung, Taewoo Ryu, K.H. Lee, and Doheon Lee. Inferring gene regulatory networks from microarray time series data using transfer entropy. In *Twentieth IEEE International Symposium on Computer-Based Medical Systems*, pages 383–388, 2007.
- 42** R.E. Ulanowicz, C. Bondavalli, and M.S. Egnotovitch. Network analysis of trophic dynamics in South Florida ecosystem, FY 97: The Florida Bay ecosystem. Technical report, Chesapeake Biological Laboratory, Solomons, MD 20688-0038 USA, 1998. [UMCES]CBL 98-123.
- 43** R.E. Ulanowicz, J.J. Heymans, and M.S. Egnotovitch. Network analysis of trophic dynamics in South Florida ecosystems, FY 99: The graminoid ecosystem. Technical report, Chesapeake Biological Laboratory, Solomons, MD 20688-0038 USA, 2000. [UMCES] CBL 00-0176.
- 44** Robert Ulanowicz. Circumscribed complexity in ecological networks. In Abbe Mowshowitz and Matthias Dehmer, editors, *Advances in Network Complexity*. Wiley, 2012.
- 45** D. J. Watts and S. H. Strogatz. Collective dynamics of ‘small-world’ networks. *Nature*, 393(6684):409–10, 1998.
- 46** M Wibrat, B Rahm, M Rieder, M Lindner, R Vicente, and J Kaiser. Transfer entropy in magnetoencephalographic data: quantifying information flow in cortical and cerebellar networks. *Prog Biophys Mol Biol.*, 105:80–97, 2011.
- 47** Thomas Wilhelm and Jens Hollunder. Information theoretic description of networks. *Physica A*, 385:385–396, 2007.

